AMENDMENTS TO THE CLAIMS

1. (Previously Presented) A method of rendering a page, the method comprising the computer-implemented steps of:
in response to receiving a request to display the page, performing the steps of:
determining that the page is associated with a page parameter;
inspecting a mapping to determine that the page parameter is mapped to a portlet parameter of a portlet that generates a component of the page that is based, at least in part, on the portlet parameter;
passing a value associated with the page parameter as a value of the portlet parameter to the portlet that generates the component of the page;
the portlet generating the component based upon the value associated with the portlet parameter; and
inserting the component that was generated by the portlet into the page.

2. (Previously Presented) The method of Claim 1, further comprising the step of mapping the page parameter, wherein mapping the page parameter comprises the steps of:
mapping the page parameter to a second portlet parameter associated with a second component of the page; and
passing the value associated with the page parameter as the value of the second portlet parameter to a second portlet that generates the second component.

3. (Previously Presented) The method of Claim 1, further comprising the steps of:
establishing a plurality of page parameters for the page; and
mapping the plurality of page parameters to a plurality of portlet parameters associated with the component of the page;
wherein the step of inspecting the mapping further comprises the step of inspecting the mapping to determine which page parameters of the plurality of page parameters are mapped to each of the plurality of portlet parameters;

wherein the step of passing the value further comprises the step of passing, based on the mapping, values associated with the plurality of page parameters as the values of the plurality of portlet parameters to the portlet that generates the component; and

wherein the step of the portlet generating the component further comprises the step of the portlet generating the component based upon the values associated with the plurality of portlet parameters.

4. (Previously Presented) The method of Claim 1, further comprising the step of mapping the page parameter to the portlet parameter associated with the component of the page without mapping the page parameter to portlet parameters associated with any other components of the page.

5. (Previously Presented) The method of Claim 1, further comprising the steps of mapping the page parameter to the portlet parameter and mapping a second page parameter to a second portlet parameter of the portlet that generates the component of the page.

6. (Previously Presented) The method of Claim 1, further comprising the step of establishing for the page parameter a default value, and wherein the step of passing the value associated with the page parameter further comprises the step of passing the default value as the value of the portlet parameter to the portlet that generates the component.

7. (Original) The method of Claim 1, wherein the request to display the page includes a URL and the URL includes the value associated with the page parameter, and wherein the step of passing the value associated with the page parameter is performed by passing the value contained in the URL as the value of the portlet parameter.

8. (Previously Presented) The method of Claim 1, further comprising the steps of:

presenting to a user a user interface for customizing the page;

in response to the user interacting with the user interface, obtaining a user specified value for the page parameter; and

wherein the step of passing the value associated with the page parameter is performed by passing the user specified value as the value of the portlet parameter to the portlet that generates the component.

9. (Previously Presented) The method of Claim 1, wherein a plurality of values are specified for the page parameter and wherein:

the method further comprises the step of determining a selected value from the plurality of values based on an override hierarchy; and

the step of passing further comprises the step of passing the selected value as the value of the portlet parameter to the portlet that generates the component.

10. (Previously Presented) The method of Claim 9, wherein the plurality of values includes a URL page parameter value and a customized page parameter value and the override hierarchy specifies that the URL page parameter value is the selected value.

11. (Previously Presented) The method of Claim 9, wherein the plurality of values includes a default page parameter value and a customized page parameter value and the override hierarchy specifies that the customized page parameter value is the selected value.

12. (Previously Presented) The method of Claim 9, wherein the plurality of values includes a default page parameter value and a portlet specified value and the override hierarchy specifies that the default page parameter value is the selected value.

13. (Original) The method of Claim 1, further comprising the step of presenting to a page designer a user interface for specifying the mapping between the page parameter and the portlet parameter.

14. (Previously Presented) The method of Claim 1, further comprising the step of registering the portlet with a portal repository, wherein the process of registering the portlet causes data associated with the portlet to be stored in the portal repository.

15. (Previously Presented) The method of Claim 14, wherein the data associated with the portlet is communicated to the portal repository as an XML document.

16. (Previously Presented) The method of Claim 1, further comprising the step of receiving input from a page designer, through a user interface, to create the mapping between the portlet parameter and the page parameter.

17. (Previously Presented) The method of Claim 1, wherein the value associated with the page parameter is stored in memory and wherein:
the method further comprises the step of retrieving the stored value; and
the step of the portlet generating the component further comprises the step of the portlet generating the component based upon the retrieved value.

18. (Currently Amended) A method comprising the computer-implemented steps of:
in response to a user manipulating a component associated with a page, a portlet that ~~generates~~ previously generated the component generating a particular event;
the portlet passing data that represents the particular event to logic associated with the page;
inspecting a first mapping that maps events to actions and event output parameters to page parameters;
determining, based on the first mapping and the passed data, an action to perform in response to the particular event;
inspecting the first mapping to determine that an event output parameter associated with the particular event is mapped to a page parameter; and

causing the action to be performed, wherein causing the action to be performed comprises passing a value of the event output parameter as the value of the page parameter.

19. (Previously Presented) The method of Claim 18, wherein:
the page is a first page and the page parameter is associated with a second page; and
the step of causing the action to be performed further comprises the step of passing the value of the page parameter to logic responsible for rendering the second page.

20. (Previously Presented) The method of Claim 18, wherein the step of causing the action to be performed further comprises the step of generating a request that specifies a URL, wherein the value of the page parameter is included in the URL.

21. (Original) The method of Claim 20, wherein:
the step of generating the request further comprises the step of generating a request for executable code; and
the step of causing the action to be performed further comprises the step of invoking the executable code.

22. (Original) The method of Claim 21, wherein the executable code is a web service.

23. (Previously Presented) The method of Claim 18, wherein:
the action comprises rendering a second page, wherein the page parameter is associated with the second page, and wherein rendering the second page comprises the steps of:
inspecting a second mapping to determine that the page parameter is mapped to a portlet parameter of a second portlet that generates a second component of the second page that is based, at least in part, on the portlet parameter;

passing the value of the page parameter as the value of the portlet parameter to the second portlet;

the second portlet generating the second component based upon the value associated with the portlet parameter; and

inserting the second component that was generated by the second portlet into the second page.

24. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 1.

25. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 2.

26. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 3.

27. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 4.

28. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 5.

29. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 6.

30. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 7.

31. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 8.

32. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 9.

33. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 10.

34. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 11.

35. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 12.

36. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 13.

37. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 14.

38. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 15.

39. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 16.

40. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 17.

41. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 18.

42. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 19.

43. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 20.

44. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 21.

45. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 22.

46. (Original) A computer-readable medium carrying one or more sequences of instructions which, when executed by one or more processors, causes the one or more processors to perform the method recited in Claim 23.